IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 12, No. 2, June 2023, pp. 627~640
ISSN: 2252-8938, DOI: 10.11591/ijai.v12.i2.pp627-640

A hybrid approach for face recognition using a convolutional neural network combined with feature extraction techniques

Hicham Benradi, Ahmed Chater, Abdelali Lasfar
High School of Technology Salé, Mohammadia School of Engineering, Systems Analysis and Information Processing and Industrial Management Laboratory, Mohammed V University, Rabat, Morocco

Article Info

Article history:

Received Feb 6, 2022
Revised Sep 27, 2022
Accepted Oct 27, 2022

Keywords:

Activation function
Convolutional neural network
Deep learning
Facial recognition
Optimization algorithms
Scale invariant feature transform+convolutional neural network

ABSTRACT

Facial recognition technology has been used in many fields such as security, biometric identification, robotics, video surveillance, health, and commerce due to its ease of implementation and minimal data processing time. However, this technology is influenced by the presence of variations such as pose, lighting, or occlusion. In this paper, we propose a new approach to improve the accuracy rate of face recognition in the presence of variation or occlusion, by combining feature extraction with a histogram of oriented gradient (HOG), scale invariant feature transform (SIFT), Gabor, and the Canny contour detector techniques, as well as a convolutional neural network (CNN) architecture, tested with several combinations of the activation function used (Softmax and Segmoïd) and the optimization algorithm used during training (adam, Adamax, RMSprop, and stochastic gradient descent (SGD)). For this, a preprocessing was performed on two databases of our database of faces (ORL) and Sheffield faces used, then we perform a feature extraction operation with the mentioned techniques and then pass them to our used CNN architecture. The results of our simulations show a high performance of the SIFT+CNN combination, in the case of the presence of variations with an accuracy rate up to 100%.

Corresponding Author:

Hicham Benradi
Laboratory of Systems Analysis and Information Processing and Industrial Management Laboratory
High School of Technology Salé, Mohammadia School of Engineering, University Mohammed V
Rabat, Morocco
Email: benradi.hicham@gmail.com

1. INTRODUCTION

Facial recognition is a technology that belongs to the field of computer vision. It is used in several fields such as crime detection from video surveillance [1], security [2], drowsiness and fatigue detection [3], biometric security [4], and also in robotics [5]. It aims at identifying and authenticating a person from an image or a video sequence. This authentication is done from a process that starts with the detection and extraction of the face from an image or a video sequence, then the extracted face is processed by feature extraction techniques [6], [7]. This operation consists of performing a mathematical transformation calculated on all the pixels of an image, allowing it to identify the visual properties of an image so that they can be used for further processing. This operation can be performed using several techniques such as the histogram of oriented gradient (HOG) descriptor [8], scale invariant feature transform (SIFT) which is a point of interest detector, Gabor filter [9] which is a dedicated convolution filter for texture analysis, and CANNY edge detector which is an image contour detector. The final phase of this process aims at authentication by performing a comparison with other images stored in a database [10], using classifiers such as support vector machine (SVM) [11], k-nearest neighbors (KNN) [12], or principal component analysis (PCA) [13].

Recently, several techniques have been implemented [14]–[17] , aiming at identifying an individual from an image with maximum accuracy. However, the efficiency of a facial recognition system remains ineffective in case of variations in the processed image such as pose, lighting, pre and sence of occlusion. Recently, most of the work in face recognition uses the deep learning technique which is based on the architecture of convolutional neural networks (CNN) inspired by biological neural networks, because of their robustness in terms of analysis and feature extraction thanks to their deep architecture, allowing to perform a very large number of computations on a single region of an image. They are used in several domains, such as medicine [18]–[20], agriculture [21]–[23], and economics [24].

A CNN is an architecture that consists of several layers and parameters. There are two main parts in this architecture, which are the feature extraction part and the classification part. The first part mainly uses convolution layers to extract features and form a feature map and a pooling layer which reduces the dimension of the feature map to reduce the computation rate. In the second part, a fully connected layer is used to assign each image to a specific class that is suitable for it, based on the result of the first part, and another layer called dropout is used to avoid overlearning our model in the training data set. It randomly drops several neurons during the model training to reduce the size of the model. It also uses an activation function that selects the most relevant variables to pass to the next neuron. There are several activation functions such as Relu Softmax, and Sigmoide, and each function is used according to the context of the classification sought, either a binary or a multi-class classification. After the design of the CNN, the architecture comes the last part which is the training of the model. This part contains an important parameter, the optimization algorithm used, which reduces the error rate. There are several types of optimizations algorithms, among which are d stochastic gradient descent (SGD).

In this paper, we propose a hybrid approach based on the best feature extraction algorithm with different face variations, which is associated with the CNN architect modified according to the Softmax and Sigmoide activations function that allocancelingcel the negative values and speed up the processing time and finally evaluate by the following optimization techniques: Adam, Adamax, RMSprop, and SGD. To evaluate our technique, we use two databases which are: Our database of faces (ORL) and Sheffield [25], with different percentages of the base tested and trained, which present different variations (illumination, contrast, occlusion, rotation). The results of our simulations allowed us to give good results in terms of accuracy rate which reached 100% with different face variations.

2. RELATED WORK

CNN has attracted the attention of many researchers in the field of face recognition due to its enormous capacity in terms of accuracy rate. Research works have been developed based on deep learning, the authors in [26] proposed a face recognition system in the case of the presence of occlusion or noisy faces that is based on deep learning using a deep neural network (DNN), for this the features are extracted in a cascade from the images separately is then processed to select the most relevant then these are used by a DNN for a classification. Experimental results showed that this method achieved an accuracy of 92.3%. Another method has been proposed for face recognition under unfavorable conditions (difficult lighting, blur, and low resolution) by [27] which uses CNNs to project the covariance matrices of Gabor waves into a feature vector of the Euclidean space. This method effectively extracts fine features from an image and has been shown to perform better than DNN. Another face recognition method was developed for use in a Big data environment by [28] who optimized a face recognition algorithm that combines two feature extraction techniques which are local binary pattern (LBP) algorithm and two-dimensional principle component analysis (2DPCA) these features are subsequently merged to pass them to a CNN as input data. In the context of big data, the accuracy of this technique could exceed 90%. The use of the linear discriminant analysis (LDA) technique to generate a set of one-dimensional facial features from an original image dataset for training the classifier of a one-dimensional deep convolutional neural network (1D-DCNN) was used in [29]. This method demonstrated a high accuracy performance that reached 100% accuracy.

3. METHODOLOGY

The proposed approach is based on four essential steps. First, a pre-processing of the data used. Then a feature extraction operation is performed using techniques (HOG, SIFT, GABOR and CANNY). Then, using a convolutional neural network architecture (CNN), we train our classification model. Finally, a calculation of the accuracy rate will be performed.

3.1. Preprocessing of the data used

3.1.1. Database used

ORL dataset: ORL is a database of faces that contains 400 images in total distributed over 40 individuals, i.e. 10 images per individual. The conditions under which these images were taken differ either

by the details of the face (with or without glasses) or by the variance of the lighting or the facial expression (smiling or not, eyes open or closed). All the images are in gray level with a size of 92x112 pixels. Figure 1 shows some images belonging to the database.

Sheffield dataset: Sheffield is a database of faces of 20 individuals and contains a total of 564 gray level images of identical size of 220X220 pixels [25]. Each individual is represented by poses ranging from profile to full face view taken under different conditions with variance either by gender, race, or appearance. Figure 2 shows some images belonging to the database.

Figure 1. Example of images with variance from the ORL database

Figure 2. Layout from profile to front view from Sheffield database

3.1.2. Data preparations

A pre-processing is performed on all the images used. It aims to label each image with a corresponding label for each set of images of the same individual, then transform them into the gray level, then resizes them to an identical size which is 48×48 pixels to convert them into an array of pixel values and reconvert them into float and finally to form two groups of data which are the first called Train for the training part of the model and the second test which will be used to test the effectiveness of the model. The Figure 3 shows the final form of the preprocessing performed on the images.

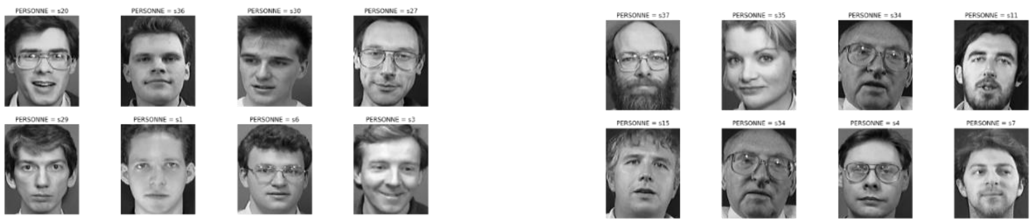


Figure 3. Preprocessing was performed on all the images of the ORL database

3.2. Feature extraction

3.2.1. Scale invariant feature transform (SIFT)

SIFT [30], is a very powerful and well-recognized point of interest detector in the field of facial recognition, it is based on the calculation of the euclidean distance between two vectors to determine if they correspond to the same points of interest in different images [31]. This technique goes through four main steps Scale-space extrema detection, localization of key points, Orientation assignment, and extraction of key point descriptors [30] detailed,

- In the first step, the different key points are identified in an image using the difference of Gaussians (DOG) from multiple Gaussian images produced at different scales from the original image where the DoGs are computed from the neighbors in the scale space.
- The second step consists of locating the different candidate keypoints based on extrema existing in the DoGs, eliminating unstable keypoints with low contrast [30].
- The third step assigns a principal orientation to each key point.
- The final phase computes a highly distinctive descriptor for each keypoint.

The Figure 4 shows an example of key point detection by the SIFT technique. This detection was performed using images from the ORL face database, with a size of 112×92 pixels. A set of parameters was used to detect its key points, defined: the number of layers in each octave is 3, the contrast threshold is a value of 0.04, the threshold for filtering the edges is a value of 10 and the value of the sigma of the Gaussian is 1.6.

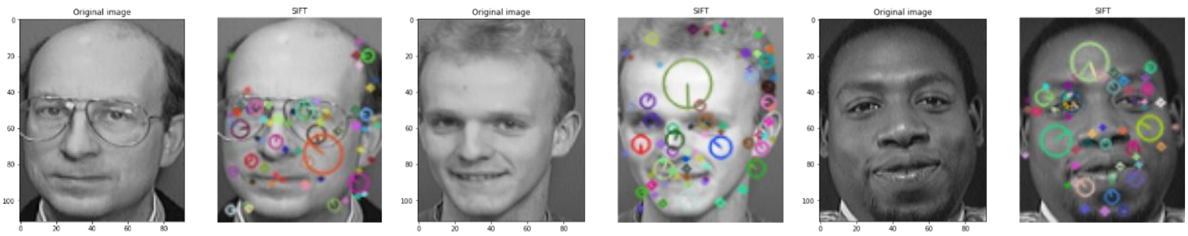


Figure 4. Detection of key points by the SIFT

3.2.2. Histogram of oriented gradient (HOG)

HOG or histogram of the oriented gradient is a descriptor used in the field of image processing. It was proposed by Dalal and Triggs in [8] and it aims to represent the appearance and local shape of an object in the image by a distribution of the intensity of the gradient. To do this, the image must be resized to a size of 64×128 pixels because it uses a detection window of the same size, then divide the image into small cells and then calculate each cell the histogram of the gradient directions by calculating the magnitude using the following formula,

$$||\nabla f|| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \quad (1)$$

Where∂x: the value of the gradient in the horizontal direction
∂y: the value of the gradient in the vertical direction
Then the orientation of the gradient is calculated by

$$\theta = tan^{-1}\left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x}\right) \quad (2)$$

And at the end, all these histograms will be combined to form a HOG descriptor.

Figure 5 shows an example of edge detection using the HOG technique. This detection was performed using images from the ORL face database, with a size of 112×92 pixels. For this, the number of orientation bins is 9, the cell size is 8×8 pixels, and several cells in each block for histogram normalization (2.2) :

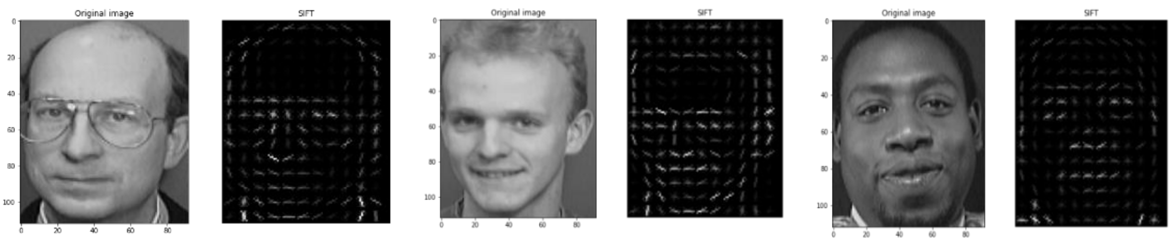


Figure 5. Descriptor detection by the HOG

3.2.3. The Canny contour detector

Canny is an image processing algorithm whose goal is to detect the contour of an image [32]. It allows to reduce the amount of data to be processed and this by extracting useful structural information. This detection is achieved by following these steps,

1) The first step is to remove the noise using a Gaussian filter because the presence of noise in an image influences the detection of the contour.

2) Then a search of the intensity gradient of the image is performed using a Sobel filter this is done in both directions of the image (horizontal Gx and vertical Gy) and from these two images the gradient and the direction of the edges for each pixel are identified using the formula,

$$Edge\_Gradient(G) = \sqrt{G_x^2 + G_y^2} \tag{3}$$

$$Angle(\theta) = tan^{-1}\left(\frac{G_y}{G_x}\right) \tag{4}$$

Where: G_x: is the first derivative in the horizontal direction.
G_y: is the first derivative in the vertical direction.

3) By analyzing the entire image, we eliminate the unwanted pixels that do not correspond to an edge. This operation is performed by comparing a pixel with its neighborhood in the direction of the gradient. Finally checked if it is a local maximum then it is an edge otherwise it is eliminated. Figure 6 is a graphical presentation of the edge selection operation. The first case Figure 6(a) shows how the CANNY detector proceeds to eliminate a pixel. The second case shows Figure 6(b) how CANNY selects an edge.

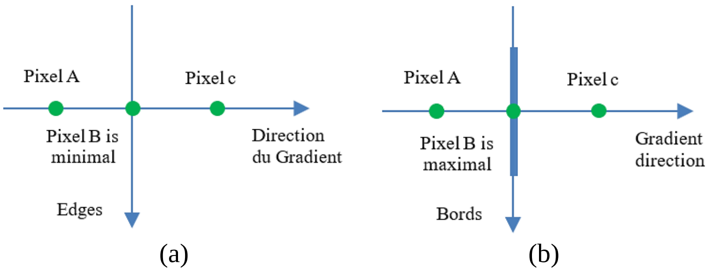


Figure 6. Determination of an edge (a) Case of elimination of minimal pixel B in the neighborhood (pixels A and C) not corresponding to an edge and (b) Maximum pixel B in the neighborhood (pixels A and C) corresponding to an edge

4) The last step is to select the best edges using the hysteresis threshold. To do this we need two threshold values, minval and maxval, then all the edges with an intensity gradient greater than maxval are considered as edges and in the case where the intensity gradient is less than minval is eliminated because it is not an edge. In the case where the intensity gradient is located between the two values minval and maxval, checking if it is connected to another pixel higher than maxval, in this case, it is considered as an edge.

The Figure 7 shows an example of edge detection using the CANNY technique. This detection was performed using images from the ENT face database, with a size of 112×92 pixels. For this, the threshold value of the maximum intensity gradient used is 100 and the minimum value is 200,

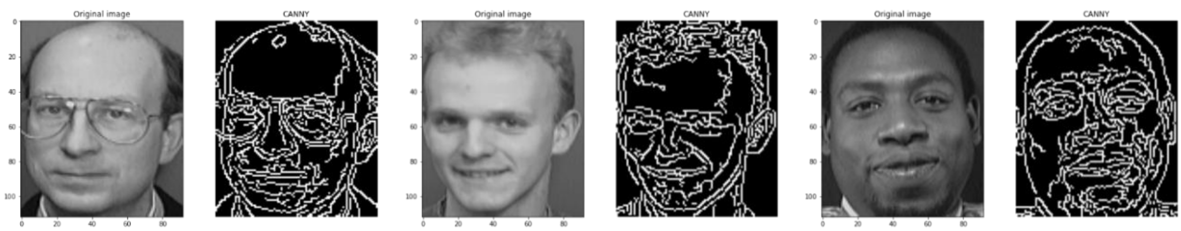


Figure 7. Descriptor detection by CANNY

3.2.4. Gabor filter

A Gabor filter is a convolution filter used in computer vision and especially image processing for texture analysis, edge detection, or feature extraction from an image [9]. They are special classes of bandpass filter because it carries out a selection of the bands of frequencies to identify among them which one it must

allow and which one must be rejected. It represents a combination of Gaussian and sinusoidal terms where the Gaussian component provides the weights and the sinusoidal component provides the direction, as the following formula shows,

$$g(x,y,\lambda,\theta,\psi,\sigma,\gamma)=\exp\left(-\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}\right)\exp\left(i\left(2\pi\frac{x'}{\lambda}+\psi\right)\right) \tag{5}$$

Where: λ: wavelength of the sinusoidal component. θ: orientation of the filter. ψ: phase shift. σ: gaussian filter. γ: spatial aspect ratio and: x' = x cos θ + y sin θ and y' = -x sin θ + y cos θ

The Gabor kernel mimics the visual cortex which means simulating using the Gabor kernel how do we recognize the texture with our eyes that can be, which round it a bank of filters that can be designed to detect the texture and extract the texture. The Figure 8 shows an example of detecting key points of images from the ORL face database, of size 112×92 pixel by Gabor. For this purpose, six filters with six main directions were used to extract the features from the images. Their angle is defined (0°, 30°, 60°, 90°, 120°, and 150°) respectively. The size of each filter is 11 pixels, the standard deviation of the Gaussian function Sigma= 1.5, and the wavelength of the sinusoidal factor Lambda= 3.

The Figure 9 summarizes the result of the feature extraction phase on the same images. For this we used the HOG, SIFT, CANNY and GABOR techniques to extract features from images. It is noted that each technique was used with a specific parameterization.

Figure 8. Descriptor detection by GABOR

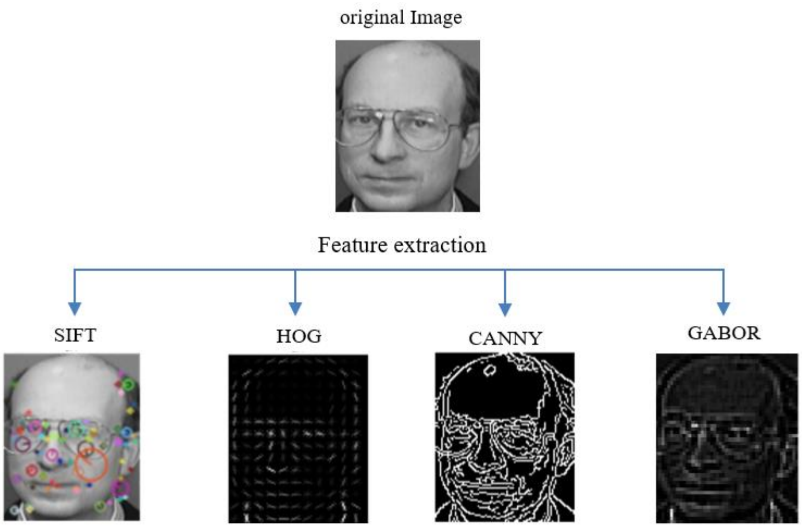


Figure 9. Feature extraction from ORL database images

3.3. Model training using a convolutional neural network CNN

This phase is devoted to the creation of our CNN. Its objective is to train our model using the Train part and to validate the model using the Test part. For this we have adopted the following architecture,

- Three convolution layers which contain respectively a filter value of 6, 16, and 64 with a kernel of 5x5 for the first two layers and 3x3 for the third and rectified linear unit (ReLU) as activation function.

– Each layer will be followed by a Maxpool layer with a value of (2.2), which will have the goal of reducing the input size to half.
– A Flatten layer to flatten all values.
– Two fully connected dense layers the first one uses ReLU as an activation function followed by a Dropout to avoid overfitting at a threshold set to 0.5 and the second one will use Softmax for the first case and Sigmoid for the second case the as activation function for classification.

The Figure 10 is a summary diagram of the architecture of our neural network developed in our method,

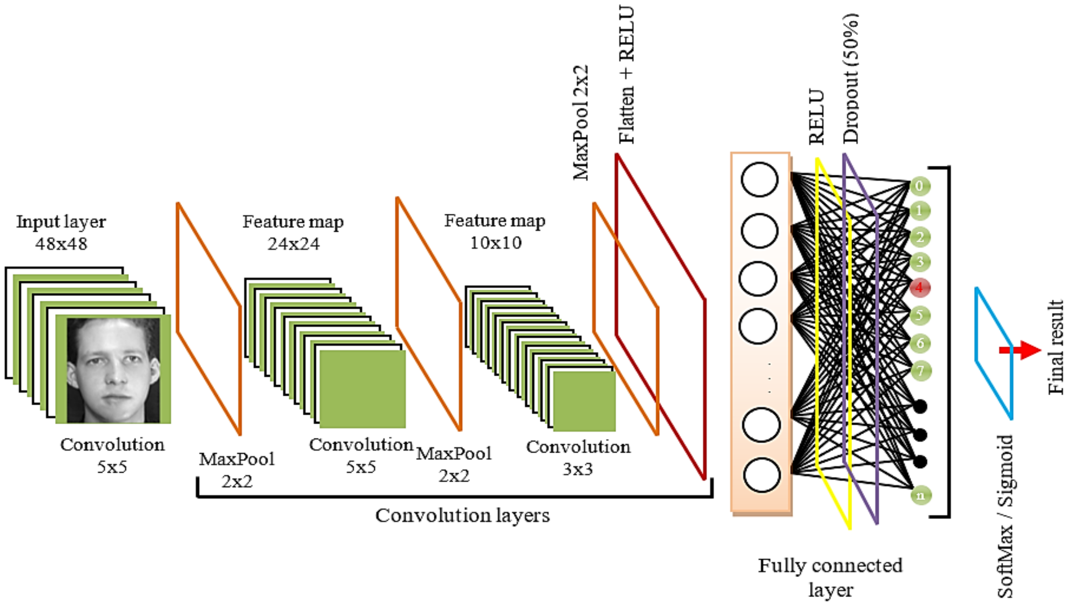


Figure 10. The architecture of the CNN neural network used

our elaborated CNN architecture has been tested with several combinations using two different activation functions Softmax and Segmoïd, in the dense classification layer, and the optimization algorithms adam, Adamax, RMSprop, and SGD, used in the training to evaluate which of these combinations gives better results. Activation functions are functions capable of spatially modifying the representation of the data, allowing it to switch from a linear to a non-linear form. For our approach, we used three types of activation functions which are :

Rectified linear unit**:** Known for its simplicity which makes it the most used among the other activation functions, the rectified linear unit or ReLU function [33] is a function that aims at determining the maximum between x and 0.

$$\begin{cases} x \text{ if } x > 0 \text{ and} \\ 0 \text{ if } x < 0 \end{cases}$$

$$\text{Fonction\_ReLU}(x) = \max(x, 0) \tag{6}$$

Softmax**:** This is a function often used by classification models for multi-class problems. It treats each vector independently of the others and allows the transformation of a real vector into a probability vector. The input axis on which Softmax is applied is defined by the axis argument.

$$\text{fonction\_Softmax}(x) = \exp(x)/\text{tf.reduce\_sum}(\exp(x)) \tag{7}$$

$$\text{fonction\_Softmax}(x) = \exp(x)/\text{sum}(\exp(xi)) \tag{8}$$

Sigmoid**:** it is a function that is used for binary classification in the case where the model will have to determine only two labels because the results are always between 0 and 1. It uses the following formula,

$$\text{fonction\_Sigmoid}(x) = 1/(1 + \exp(-x)) \tag{9}$$

during the training of the model four optimization algorithms were used which are:

Adam: proposed by Kingma & Ba in 2014 in their paper [34], Adam is an optimization algorithm based on stochastic gradient descent. Its role is to update the weights of a neural network iteratively, according to the training data. For that it uses the following formula,

$$\theta_{n+1} = \theta_n - \frac{\alpha}{\sqrt{\hat{v}_n + \varepsilon}} \hat{m}_n \tag{10}$$

Where: θ_{n+1}: weight of time n+1; θ_n: weight of time n; v_n: sum of the square of the past gradients
α: step size parameter; ϵ: a constant; m_n: the aggregate of the gradient of time n

Adamax: this is an extension of the Adam optimization algorithm. Adamax [34] updates the weights of a neural network based on the infinite norm of past gradients. For this it uses the following formula,

$$\boldsymbol{\theta_{n+1} = \theta_n - \frac{\eta}{u_n} \hat{m}_n} \tag{11}$$

Where: θ_{n+1}: weight of time n+1; θ_n: weight of time n; m_n: gradient aggregate of time n
η: 0.002; u_n: is used to denote the infinite norm constraint vt,

$$u_n = \beta_2^{\infty} v_{n-1} + (1 - \beta_2^{\infty})|g_n|^{\infty} \tag{12}$$

$$u_n = max(\beta_2 . v_{n-1}, |g_n|) \tag{13}$$

Where: β1 and β2 are default values (β1=0.9 and β2=0.999)

RMSprop: Proposed by Geoffrey Hinton, RMSpro is an optimization technique used in the training phase. It was designed for mini-batch learning as a stochastic technique to solve the gradient explosion problems using complex functions. It is based on the gradient and specifically the normalization of the gradient. The step size is modified to avoid the explosion in the case of large gradients the step size is decreased and to avoid the disappearance in the case of small gradients the step size is increased to create an equilibrium. In other words, it aims to change the learning rate according to the size of the gradients. RMSprop update rule,

$$E[g^2]_n = \beta E[g^2]_{n-1} + (1 - \beta)\left(\frac{\delta c}{\delta w}\right)^2 \tag{14}$$

$$\theta_{n+1} = \theta_n - \frac{\eta}{\sqrt{E[g^2]_n}} \frac{\delta c}{\delta w} \tag{15}$$

Where: E[g]: moving average of the square gradients
δC/δw: gradient of the cost function concerning the weight
η: learning rate; β: moving average parameter (0.9)

Stochastic gradient descent (SGD): is an optimization algorithm. It eliminates the redundant computations performed by batch gradient descent in the case of large data sets which recalculates the gradients before each parameter update and performs one update at a time using the following formula,

$$\theta = \theta - \eta . \nabla_\theta j(\theta; x^{(i)}; y^{(i)}) \tag{16}$$

Where: θ : model parameters ; η: learning rate ; ∇θJ(θ) : objective function
$x^{(i)}$: training set; label

3.4. Calculation of the accuracy rate

The last step of our method is devoted to the evaluation of the model by calculating the accuracy rate. For this, we used the categorical_crossentropy function as a loss function. It is generally used in multi-class classification and its purpose is to decide in which class among all existing classes a result from a CNN can be assigned using the following formula,

$$Loss = -\sum_{i=1}^{output\ size} y_i . \log \hat{y}_i \tag{17}$$

Where: output size: number of scalar values in the model output
$\hat{y}_i$: scalar value in the output of the model; corresponding target value
The following diagram as shown in Figure 11 summarizes the different steps of our approach,

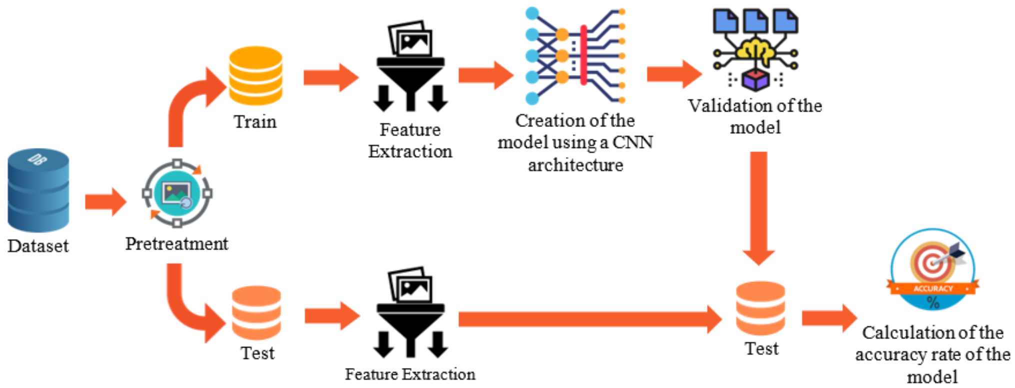


Figure 11. Summary diagram of the steps of our method

4. RESULTS AND DISCUSSION

The results of our simulations were performed on two databases ORL and Sheffield. The purpose of these simulations is to evaluate the accuracy rate for each database. All models were trained using 150 epochs and a batch size of 40 for the ORL database and 58 for the Sheffield database.

4.1. Evaluation of the proposed method using the ORL database

The results of our simulations performed on the Test part of the ORL database using the Softmax activation function gave satisfactory results in terms of accuracy rate in the case of using Adam, Adamax, and RMSprop optimizers with a value higher than 80% for the feature extraction techniques HOG, SIFT, Gabor and Canny. On the other hand, using the SGD optimization algorithm, the accuracy gave less compared to the others. As we can see from the Table 1 we can say that the best accuracy is obtained using the Adam optimizer with an accuracy rate varying between 80% as the minimum value in the case of Canny and 93.75% as the maximum value in the case of Gabor.In the case of using the Sigmoid activation function, the results obtained by the HOG technique are the lowest compared to the other feature extraction techniques used, especially in the case of using SGD as the optimization algorithm. The reading made on the Table 1, which represents graphically the totality of our simulation results, shows that when using the combination of SIFT with the CNN the results obtained are satisfactory and stable whatever the combination (activation function/optimization algorithm) used, with an accuracy rate that reached 92.5%.

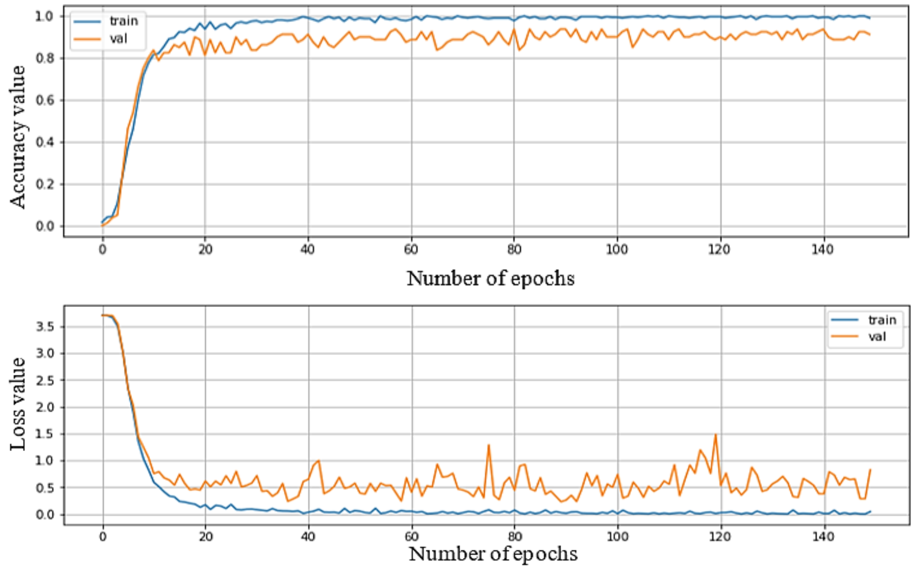
The results of our simulations are the Figure 12, in terms of accuracy and loss rate of the SIFT+CNN method, Figure 12(a) presents the Softmax/RMSprop combination use case and Figure 12(b) presents the Sigmoid/Adamax combination use case. The accuracy on the training set reaches a value of 100% and a value of 91.25% on the test set for validation in the Softmax/RMSprop case and the loss function reaches a minimum value of 0.0679 on the training set and 0.187 on the validation set. In the case of using the Sigmoid/Adamx combination, a value of 98.12% was achieved on the training set and 92.5% for the validation set. For the loss function, it reached a minimum value of 0.0013 on the training set and 0.0047 on the validation set.

4.2. Evaluation of the proposed method using the Sheffield database

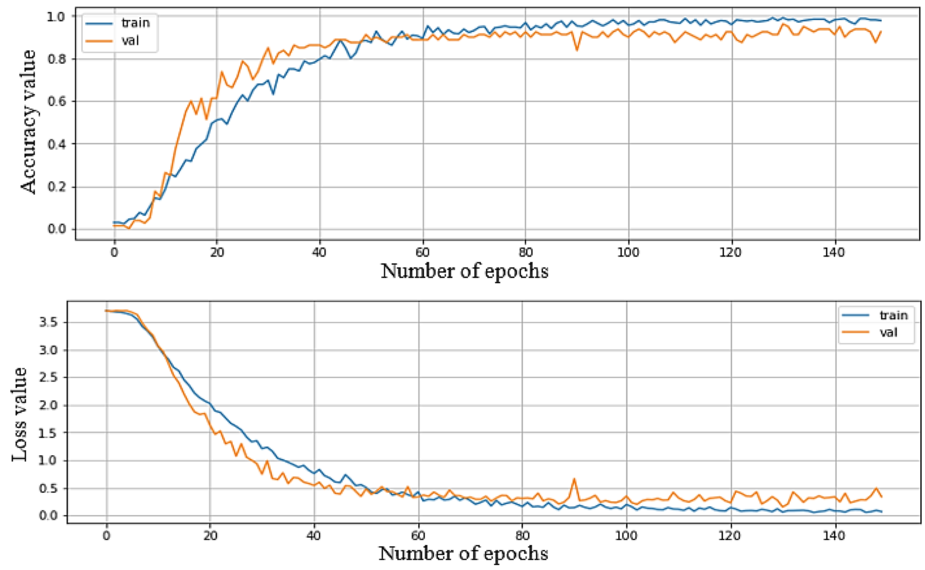
The results of our simulation performed using the Softmax activation function in the function of optimization algorithms (Adam, Adamax, RMSprop, and SGD), allowed us to record good results in terms of accuracy rate. In the case of using the three optimization algorithms adam, Adamax and RMSprop, the accuracy rate exceeded the value of 97.39%. It is also noted that the combination SIFT+CNN has recorded the highest accuracy rates (adam = 100%, Adamx = 99.13%, RMSprop = 100% and SGD=98.26%).

Table 1. The performance of accuracy rate in (%) according to the ORL face database

	Fonction d'activation Softmax				Fonction d'activation Sigmoid			
Optimisateur	Adam	Adamax	RMSprop	SGD	Adam	Adamax	RMSprop	SGD
HOG + CNN	91.25	92.5	91.25	0	5	91.25	97.5	0
LBP + CNN	68.75	75	77.45	0	45	63.75	3.75	0
SIFT + CNN	88.75	88.75	91.25	86.25	88.75	92.5	91.25	92.5
GABOR + CNN	93.75	88.75	90	88.75	90	87.5	90	92.5
CANNY + CNN	80	86.25	87.5	83.75	82.5	83.75	87.5	87.5



(a)



(b)

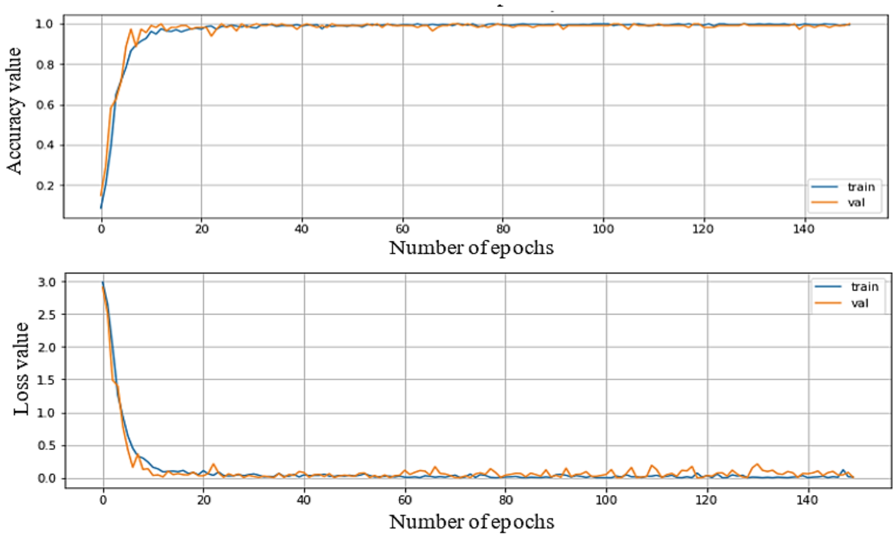
Figure 12. Accuracy and loss rates as a function of the number of epochs according to the SIFT +CNN technique, (a) Softmax+RMSprop, and (b) Sigmoid+Adamax

For the case of using the Sigmoid activation function, the results obtained show that the accuracy rate decreases when using the SGD optimization algorithm. On the other hand, using other algorithms gives satisfactory results, especially when using the Adamax algorithm, because all the rates recorded are high, varying between 96.52% and 99.13%. The results as shown in the Table 2, shows a remarkable performance of the feature extraction techniques Sift, Gabor, and Canny. On the other hand, the rates recorded by the Hog techniques are less satisfactory.

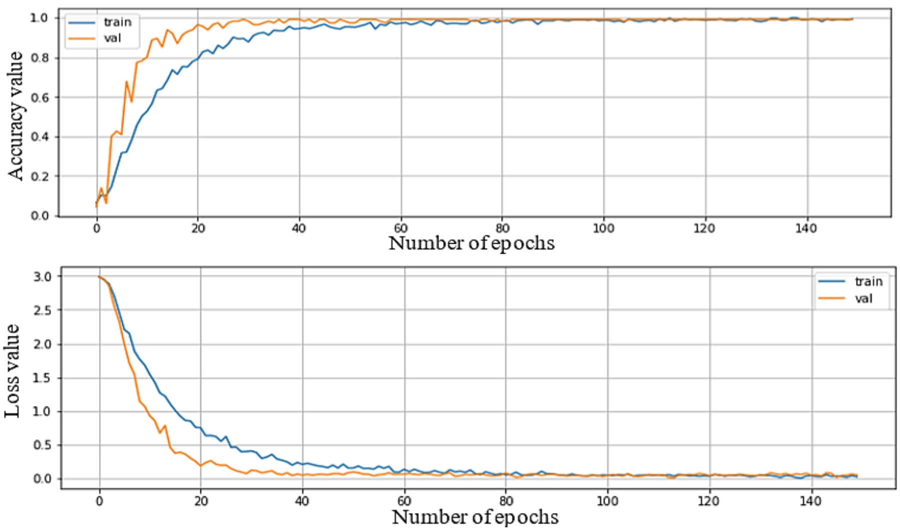
Table 2. The performance of accuracy rate in (%) according to the Sheffield face database

	Fonction d'activation Softmax				Fonction d'activation Sigmoid			
Optimisateur	adam	Adamax	RMSprop	SGD	adam	Adamax	RMSprop	SGD
HOG + CNN	97.39	99.13	97.39	92.17	9.56	98.26	99.13	10.43
LBP + CNN	89.56	87.82	97.39	10.43	91.3	96.52	6.08	4.34
SIFT + CNN	100	99.13	100	98.26	98.26	99.13	99.13	97.39
GABOR + CNN	98.26	99.13	99.13	98.26	98.26	99.13	99.13	97.39
CANNY + CNN	98.26	97.4	99.13	96.52	97.4	96.52	98.26	98.26

The results of our simulations are shown in the Figure 13 in appendix, in terms of accuracy and loss rates of the SIFT+CNN method, Figure 13(a) shows the use case of Softmax/RMSprop combination and Figure 13(b) shows the use case of Sigmoid/Adamax combination. The accuracy on the training set reaches a value of 99.56% and a value of 100% on the test set for validation in the Softmax/RMSprop case and the loss function reaches a minimum value of 0.0090 on the training set and 0.0039 on the validation set. In the case of using the Sigmoid/Adamx combination, a value of 99.34% was achieved on the training set and 99.13% for the validation set. For the loss function, it reached a minimum value of 0.0258 on the training set and 0.0511 on the validation set.



(a)



(b)

Figure 13. Accuracy and loss rates as a function of the number of epochs according to the SIFT +CNN technique, (a) Softmax+RMSprop, and (b) Sigmoid+Adamax

To better evaluate the performance of our approach, a comparative study with other existing approaches was performed. The Table 3 summarizes the accuracy rate results of our approach and other approaches applied on the same two datasets ORL and Sheffield. It results that our approach is competitive and achieves a better result in terms of recognition rate with the presence of different variations.

Table 3. Comparison of recognition rate of the proposed method with another existing approach

ORL dataset		Sheffield dataset	
Method	Accuracy (%)	Method	Accuracy (%)
NSST [15]	99.32	SURF+SVM [35]	97.87
LBP+CNN [36]	100	Parameterless SLPP [37]	95.6
DNNs [38]	99.07	2DJLNDA + CNN [39]	89.87
PCA + SVM [40]	98.75	V-LGS + LSAD [41]	95
SIFT [42]	91.2	UFSELM+L2.1 [43]	76.89
(MF_GF_HE)_PCA_ MultiSVMs [44]	91.6	AS-LRC [45]	85.42
Proposed method	100	Proposed method	100

5. CONCLUSION

Facial recognition is a field of artificial intelligence that aims to identify individuals from an image. It is a complicated task, for applications such as the identification of individuals by video surveillance when variations or occlusions are present. In this paper, we propose a hybrid approach based on the best feature extraction algorithm with different face variations, which is associated with the CNN architect modified according to the Softmax and Sigmoide activation function that allows to cancel the negative values and to speed up the processing time, and finally evaluated by the following optimization techniques: adam, Adamax, RMSprop and SGD.The results obtained showed a remarkable performance when using SIFT+CNN with an accuracy rate of up to 100%. We also note that when using the Softmax activation function and the Adam, Adamax, and RMSprop optimization algorithms, the results are satisfactory in terms of accuracy rate. This approach can be used in the field of security by video surveillance to identify individuals in case of the presence of variations. It is noted that testing on other databases representing new cases of variations is important to better evaluate our proposed model.

REFERENCES

[1] K. K. Kumar, Y. Kasiviswanadham, D. V. S. N. V. Indira, P. Priyanka palesetti, and Ch. V. Bhargavi, "Criminal face identification system using deep learning algorithm multi-task cascade neural network (MTCNN)," *Materials Today: Proceedings*, 2021, doi: 10.1016/j.matpr.2021.06.373.

[2] P. Wang, P. Wang, and E. Fan, "Violence detection and face recognition based on deep learning," *Pattern Recognition Letters*, vol. 142, pp. 20–24, 2021, doi: 10.1016/j.patrec.2020.11.018.

[3] N. N. A. Mangshor, I. A. A. Majid, S. Ibrahim, and N. Sabri, "A real-time drowsiness and fatigue recognition using support vector machine," *IAES International Journal of Artificial Intelligence*, vol. 9, no. 4, pp. 584–590, 2020, doi: 10.11591/ijai.v9.i4.pp584-590.

[4] E. Maiorana, "Deep learning for EEG-based biometric recognition," *Neurocomputing*, vol. 410, pp. 374–386, 2020, doi: 10.1016/j.neucom.2020.06.009.

[5] E. Yaghoubi, D. Borza, J. Neves, A. Kumar, and H. Proença, "An attention-based deep learning model for multiple pedestrian attributes recognition," *Image and Vision Computing*, vol. 102, 2020, doi: 10.1016/j.imavis.2020.103981.

[6] V. Vijayan and P. Kp, "A comparative analysis of RootSIFT and SIFT methods for drowsy features extraction," *Procedia Computer Science*, vol. 171, pp. 436–445, 2020, doi: 10.1016/j.procs.2020.04.046.

[7] A. Chater and A. Lasfar, "New approach to the identification of the easy expression recognition system by robust techniques (SIFT, PCA-SIFT, ASIFT and SURF)," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 18, no. 2, pp. 695–704, 2020, doi: 10.12928/TELKOMNIKA.V18I2.13726.

[8] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005*, vol. I, pp. 886–893, 2005, doi: 10.1109/CVPR.2005.177.

[9] S. Fischer, G. Cristóbal, and R. Redondo, "Sparse overcomplete gabor wavelet representation based on local competitions," *IEEE Transactions on Image Processing*, vol. 15, no. 2, pp. 265–272, 2006, doi: 10.1109/TIP.2005.860614.

[10] A. Chater and A. Lasfar, "Comparison of robust methods for extracting descriptors and facial matching," *2019 International Conference on Wireless Technologies, Embedded and Intelligent Systems, WITS 2019*, 2019, doi: 10.1109/WITS.2019.8723858.

[11] L. Shi, X. Wang, and Y. Shen, "Research on 3D face recognition method based on LBP and SVM," *Optik*, vol. 220, 2020, doi: 10.1016/j.ijleo.2020.165157.

[12] M. Sowmiya Manoj and S. Arulselvi, "Palm print identification and classification using KNN algorithm," *Materials Today: Proceedings*, 2021, doi: 10.1016/j.matpr.2021.01.804.

[13] Y. Ren, X. Xu, G. Feng, and X. Zhang, "Non-Interactive and secure outsourcing of PCA-Based face recognition," *Computers and Security*, vol. 110, 2021, doi: 10.1016/j.cose.2021.102416.

[14] A. H. Thary Al-Ghrairi, A. Abdulwahhab Mohammed, and E. Z. Sameen, "Face detection and recognition with 180 degree rotation based on principal component analysis algorithm," *IJ-AI*, vol. 11, no. 2, p. 593, Jun. 2022, doi: 10.11591/ijai.v11.i2.pp593-602.

[15] Y. S and N. Purnachand, “Convolutional neural network-based face recognition using non-subsampled shearlet transform and histogram of local feature descriptors,” *IJ-AI*, vol. 10, no. 4, p. 1079, Dec. 2021, doi: 10.11591/ijai.v10.i4.pp1079-1090.

[16] A. Chater, H. Benradi, and A. Lasfar, “Method of optimization of the fundamental matrix by technique speeded up robust features application of different stress images,” *IJECE*, vol. 12, no. 2, p. 1429, Apr. 2022, doi: 10.11591/ijece.v12i2.pp1429-1436.

[17] B. Hicham, C. Ahmed, and L. Abdelali, “Face recognition method combining SVM machine learning and scale invariant feature transform,” *E3S Web Conf.*, vol. 351, p. 01033, 2022, doi: 10.1051/e3sconf/202235101033.

[18] W. Chen *et al.*, “Deep diagnostic agent forest (DDAF): A deep learning pathogen recognition system for pneumonia based on CT,” *Computers in Biology and Medicine*, vol. 141, 2022, doi: 10.1016/j.compbiomed.2021.105143.

[19] A. Salehzadeh, A. P. Calitz, and J. Greyling, “Human activity recognition using deep electroencephalography learning,” *Biomedical Signal Processing and Control*, vol. 62, 2020, doi: 10.1016/j.bspc.2020.102094.

[20] M. Polsinelli, L. Cinque, and G. Placidi, “A light CNN for detecting COVID-19 from CT scans of the chest,” *Pattern Recognition Letters*, vol. 140, pp. 95–100, 2020, doi: 10.1016/j.patrec.2020.10.001.

[21] F. Jiang, Y. Lu, Y. Chen, D. Cai, and G. Li, “Image recognition of four rice leaf diseases based on deep learning and support vector machine,” *Computers and Electronics in Agriculture*, vol. 179, 2020, doi: 10.1016/j.compag.2020.105824.

[22] M. Agarwal, S. K. Gupta, and K. K. Biswas, “Development of efficient CNN model for Tomato crop disease identification,” *Sustainable Computing: Informatics and Systems*, vol. 28, 2020, doi: 10.1016/j.suscom.2020.100407.

[23] C. Chen, W. Zhu, and T. Norton, “Behaviour recognition of pigs and cattle: Journey from computer vision to deep learning,” *Computers and Electronics in Agriculture*, vol. 187, 2021, doi: 10.1016/j.compag.2021.106255.

[24] J. Zhao, “Efficiency of corporate debt financing based on machine learning and convolutional neural network,” *Microprocessors and Microsystems*, vol. 83, 2021, doi: 10.1016/j.micpro.2021.103998.

[25] Eds. H. Wechsler, P. J. Phillips, V. Bruce, F. F. Soulié, and T. S. Huang, “Face recognition: from theory to applications,” *Springer Berlin Heidelberg*, 1998.

[26] N. Soni, E. K. Sharma, and A. Kapoor, “Hybrid meta-heuristic algorithm based deep neural network for face recognition,” *Journal of Computational Science*, vol. 51, 2021, doi: 10.1016/j.jocs.2021.101352.

[27] C. Li, Y. Huang, W. Huang, and F. Qin, “Learning features from covariance matrix of gabor wavelet for face recognition under adverse conditions,” *Pattern Recognition*, vol. 119, 2021, doi: 10.1016/j.patcog.2021.108085.

[28] Y. Zhu and Y. Jiang, “Optimization of face recognition algorithm based on deep learning multi feature fusion driven by big data,” *Image and Vision Computing*, vol. 104, 2020, doi: 10.1016/j.imavis.2020.104023.

[29] J. Mohammed Sahan, E. I. Abbas, and Z. M. Abood, “A facial recognition using a combination of a novel one-dimension deep CNN and LDA,” *Materials Today: Proceedings*, 2021, doi: 10.1016/j.matpr.2021.07.325.

[30] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004, doi: 10.1023/B:VISI.0000029664.99615.94.

[31] A. E.-T. A. Chater, and A. Lasfar, “Face recognition using feature extraction and similarity measures,” *Technology Reports of Kansai University*, vol. 62, no. 3, p. 10, 2020.

[32] J. Canny, “A computational approach to edge detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986, doi: 10.1109/TPAMI.1986.4767851.

[33] L. Deng *et al.*, “Recent advances in deep learning for speech research at Microsoft,” *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pp. 8604–8608, 2013, doi: 10.1109/ICASSP.2013.6639345.

[34] D. P. Kingma and J. L. Ba, “Adam: A method for stochastic optimization,” *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015.

[35] B. Anand, “Face recognition using SURF features and SVM Classifier,” p. 8.

[36] J. Tang, Q. Su, B. Su, S. Fong, W. Cao, and X. Gong, “Parallel ensemble learning of convolutional neural networks and local binary patterns for face recognition,” *Computer Methods and Programs in Biomedicine*, vol. 197, p. 105622, Dec. 2020, doi: 10.1016/j.cmpb.2020.105622.

[37] F. Dornaika and A. Assoum, “Enhanced and parameterless Locality Preserving Projections for face recognition,” *Neurocomputing*, vol. 99, pp. 448–457, Jan. 2013, doi: 10.1016/j.neucom.2012.07.016.

[38] M. N. ElBedwehy, G. M. Behery, and R. Elbarougy, “Face recognition based on relative gradient magnitude strength,” *Arab J Sci Eng*, vol. 45, no. 12, Art. no. 12, Dec. 2020, doi: 10.1007/s13369-020-04538-y.

[39] J. Yu, H. Liu, and X. Zheng, “Two-dimensional joint local and nonlocal discriminant analysis-based 2D image feature extraction for deep learning,” *Neural Comput & Applic*, vol. 32, no. 10, Art. no. 10, May 2020, doi: 10.1007/s00521-019-04085-0.

[40] xinzheng Chen, L. Song, and C. Qiu, “Face recognition by feature extraction and classification,” in *2018 12th IEEE International Conference on Anti-counterfeiting, Security, and Identification (ASID)*, Xiamen, China, Nov. 2018, pp. 43–46. doi: 10.1109/ICASID.2018.8693198.

[41] R. D. Rakshit, S. C. Nath, and D. R. Kisku, “Face identification using some novel local descriptors under the influence of facial complexities,” *Expert Systems with Applications*, vol. 92, pp. 82–94, Feb. 2018, doi: 10.1016/j.eswa.2017.09.038.

[42] K. V. Arya, S. S. Rajput, and S. Upadhyay, “Noise-robust low-resolution face recognition using SIFT features,” in *Computational Intelligence: Theories, Applications and Future Directions - Volume II*, vol. 799, N. K. Verma and A. K. Ghosh, Eds. Singapore: Springer Singapore, 2019, pp. 645–655. doi: 10.1007/978-981-13-1135-2_49.

[43] J. Chen, Y. Zeng, Y. Li, and G.-B. Huang, “Unsupervised feature selection based extreme learning machine for clustering,” *Neurocomputing*, vol. 386, pp. 198–207, Apr. 2020, doi: 10.1016/j.neucom.2019.12.065.

[44] H. M. Maw, S. M. Thu, and M. T. Mon, “Face recognition based on illumination invariant techniques model,” in *2019 International Conference on Advanced Information Technologies (ICAIT)*, Yangon, Myanmar, Nov. 2019, pp. 120–125. doi: 10.1109/AITC.2019.8921027.

[45] Z. Zhang *et al.*, “Adaptive structure-constrained robust latent low-rank coding for image recovery,” in *2019 IEEE International Conference on Data Mining (ICDM)*, Beijing, China, Nov. 2019, pp. 846–855. doi: 10.1109/ICDM.2019.00095.

BIOGRAPHIES OF AUTHORS

Hicham Benradi born on October 07, 1985. He holds a bachelor's degree in Mobile Application Engineering from the Ecole Supérieure de Technologie in Salé, and a master's degree in Data Engineering and Software Development from the Faculty of Science at Mohamed V University. He is a Ph.D. student at the Mohammadia School of Engineering in Rabat. His research focuses on facial recognition methods and image processing. He can be contacted at the following email address: benradi.hicham@gmail.com.

Ahmed Chater born on December 30, 1986, in Taounate. Degree in Engineering Sciences and Techniques, specialty: Image Processing, Laboratory of Systems Analysis, Information Processing and Industrial Management, Mohamed V University Rabat (Mohammadia School). My research focuses on the Segmentation and restoration of different types of color and grayscale images. Classification and recognition of facial expressions and machine learning. He can be contacted at email: ahmedchater11@gmail.com.

Abdelali Lasfar was born on January 10, 1971 in Salé. He is a Professor of Higher Education at Mohammed V Agdal University, Salé Higher School of Technology, Morocco. His research focuses on compression methods, indexing by image content and image indexing, and knowledge extraction from images. He can be contacted at email: ali.lasfar@gmail.com.